



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«Дальневосточный федеральный университет»
(ДВФУ)

ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ (ШКОЛА)

«СОГЛАСОВАНО»

Руководитель ОП

 Артемьева И.Л.

«УТВЕРЖДАЮ»

И.о. директора департамента

 Смагин С.В.

«26» января 2022 г.



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Основы программирования на Java
Направление подготовки **09.03.04 Программная инженерия**
Форма подготовки очная

курс 3 семестр 5, 6
лекции 0 час.
практические занятия 00 час.
лабораторные работы 70 час.
в том числе с использованием МАО лек. 0 /пр. 0 /лаб. 54 час.
всего часов аудиторной нагрузки 70 час.
в том числе с использованием МАО 54 час.
самостоятельная работа 110 час.
в том числе на подготовку к экзамену 45 час.
контрольные работы (количество) не предусмотрены
курсовая работа / курсовой проект не предусмотрены
зачет 6 семестр
экзамен 5 семестр

Рабочая программа составлена в соответствии с требованиями Федерального государственного образовательного стандарта высшего образования по направлению подготовки 09.03.04 Программная инженерия, утвержденный приказом Министерства образования и науки Российской Федерации от 19 сентября 2017 г. № 920.

Рабочая программа обсуждена на заседании департамента Программной инженерии и искусственного интеллекта, протокол № 1.1 от « 24» января 2022 г.

И.о. директора департамента Программной инженерии и искусственного интеллекта С.В. Смагин, к.т.н.

Составитель: профессор департамента ПИИИИ В.М. Гриняк, д.т.н., доцент

Владивосток

2022

Оборотная сторона титульного листа РПУД

I. Рабочая программа пересмотрена на заседании кафедры:

Протокол от « ____ » _____ 20__ г. № _____

Заведующий кафедрой _____
(подпись) (И.О. Фамилия)

II. Рабочая программа пересмотрена на заседании кафедры:

Протокол от « ____ » _____ 20__ г. № _____

Заведующий кафедрой _____
(подпись) (И.О. Фамилия)

III. Рабочая программа пересмотрена на заседании кафедры:

Протокол от « ____ » _____ 20__ г. № _____

Заведующий кафедрой _____
(подпись) (И.О. Фамилия)

IV. Рабочая программа пересмотрена на заседании кафедры:

Протокол от « ____ » _____ 20__ г. № _____

Заведующий кафедрой _____
(подпись) (И.О. Фамилия)

ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Цель дисциплины - познакомить студентов с современными приёмами создания программных средств различного целевого назначения с помощью языка программирования и технологий Java.

Задачи дисциплины:

1. Развитие способности использовать знания основных концептуальных положений объектно-ориентированного и визуального направлений программирования, методов, способов и средств разработки программ в рамках этих направлений с помощью технологий Java
2. Приобретение способности использовать знания методов проектирования и производства программного продукта, принципов построения, структуры и приемов работы с инструментальными средствами, поддерживающими создание программного обеспечения на языке программирования Java
3. Освоение специфичной профессиональной терминологии на английском языке
4. Приобретение представления о проектном методе разработки программного обеспечения

Курс основан на материалах учебных курсов международной программы академического партнёрства "Академия ОРАКЛ".

Для успешного изучения дисциплины Основы программирования на Java у обучающихся должны быть сформированы следующие предварительные компетенции: таких нет.

Профессиональные компетенции выпускников и индикаторы их достижения:

Тип задач	Код и наименование профессиональной компетенции (результат освоения)	Код и наименование индикатора достижения компетенции
проектный	ПК-6. Способность использовать навыки моделирования, анализа и формальных методов конструирования программного обеспечения	ПК-6.1. демонстрирует знание основ моделирования и формальных методов конструирования программного обеспечения.
		ПК-6.2. использует формальные методы конструирования программного обеспечения.
		ПК-6.3. применяет методы формализации и моделирования программного обеспечения.

производственно-технологический	ПК-10. Способность использовать различные технологии разработки программного обеспечения	ПК-10.1. демонстрирует знание современных технологий разработки ПО (структурное, объектно-ориентированное)
		ПК-10.2. использует структурное и объектно-ориентированное проектирование при разработке ПО.
		ПК-10.3. применяет современные технологии разработки ПО
	ПК-12. Способность применять стандарты и модели жизненного цикла.	ПК-12.1. демонстрирует знание стандартов и моделей жизненного цикла ПО.
		ПК-12.2. использует модели жизненного цикла при создании ПО.
		ПК-12.3. применяет стандарты и модели жизненного цикла ПО.

Код и наименование индикатора достижения компетенции	Наименование показателя оценивания (результата обучения по дисциплине)
ПК-6.1. демонстрирует знание основ моделирования и формальных методов конструирования программного обеспечения	Знает основные модели информационных технологий, используемых при создании программ на языке высокого уровня, процесс создания программ, стандарты языков программирования,
	Умеет использовать основные модели информационных технологий при создании программ на языке высокого уровня, общие принципы построения алгоритмов, основные алгоритмические конструкции; Владеет основными моделями информационных технологий, используемых при создании программ на языке высокого уровня
ПК-6.2. использует формальные методы конструирования программного обеспечения	Знает методы использования информационных технологий при подготовке документации и текстовых редакторов при разработке программ и документации;
	Умеет использовать информационные технологии при подготовке документации и разработке программ Владеет средствами, предоставляемыми информационными технологиями для подготовки документации и разработке программ

<p>ПК-6.3. применяет методы формализации и моделирования программного обеспечения</p>	<p>Знает методы формализации и моделирования программного обеспечения;</p> <p>Умеет разрабатывать программы, использующие данные в произвольном формате</p> <p>Владеет навыками разработки формальных моделей программного обеспечения</p>
<p>ПК-10.1. демонстрирует знание современных технологий разработки ПО (структурное, объектно-ориентированное)</p>	<p>Знает ключевые понятия структурной и объектно-ориентированной парадигм программирования.</p> <p>Умеет производить анализ предметной области в рамках современных технологий разработки ПО (структурной, объектно-ориентированной)</p> <p>Владеет навыками использования современных технологий разработки ПО (структурной, объектно-ориентированной)</p>
<p>ПК-10.2. использует структурное и объектно-ориентированное проектирование при разработке ПО</p>	<p>Знает нотацию моделей структурного (DFD) и объектно-ориентированного (диаграмма классов, диаграмма перехода состояний объекта, трасса событий) проектирования</p> <p>Умеет производить функциональную декомпозицию системы, переходить от контекстной диаграммы к спецификациям процессов (при структурном проектировании); производить объектно-ориентированный анализ (переходить от объектов к классам объектов, определять отношения между классами, поведение объектов, состояния объектов и переходы от одного состояния к другому).</p> <p>Владеет навыками построения структурных и объектно-ориентированных моделей проектирования при разработке ПО</p>
<p>ПК-10.3. применяет современные технологии разработки ПО</p>	<p>Знает современные технологии разработки ПО, алгоритмы решения прикладных задач на основе типовых структур алгоритмов, методы организации сложных структур данных</p> <p>Умеет применять современные технологии разработки ПО</p> <p>Владеет синтаксисом и семантикой основных конструкций языка программирования высокого уровня, основанного на структурной и объектно-ориентированной технологии</p>

<p>ПК-12.1. демонстрирует знание стандартов и моделей жизненного цикла ПО</p>	<p>Знает содержание основных стандартов и модели жизненного цикла ПО</p> <p>Умеет работать с документацией, касающейся основных стандартов и моделей жизненного цикла ПО</p> <p>Владеет навыками работы со стандартами и различными моделями жизненного цикла ПО при создании ПО</p>
<p>ПК-12.2. использует модели жизненного цикла при создании ПО</p>	<p>Знает терминологию, нотацию различных групп моделей жизненного цикла при создании ПО</p> <p>Умеет планировать деятельность по созданию ПО в рамках выбранной модели жизненного цикла</p> <p>Владеет навыками создания ПО в рамках выбранной модели жизненного цикла для определенной предметной области.</p>
<p>ПК-12.3. применяет стандарты и модели жизненного цикла ПО</p>	<p>Знает основные группы моделей жизненного цикла к созданию программного обеспечения, их процессы, виды деятельности и задачи, основные технологические подходы, принципы структурного и объектно-ориентированного программирования, основные понятия СОМ-технологий и SOA-технологий; тенденции развития современных технологий программирования.</p> <p>Умеет выбирать модель жизненного цикла для разработки программного обеспечения, в зависимости от масштаба проекта, стабильности требований, сроков получения стабильных версий программного обеспечения, выбирать технологический подход к разработке и соответствующие средства автоматизации разработки</p> <p>Владеет навыками адаптации видов деятельности выбранной модели жизненного цикла разработки программного обеспечения к выбранному технологическому подходу и конкретизации задач по созданию соответствующих технологических артефактов.</p>

Общая трудоемкость дисциплины составляет 5 зачётных единиц (180 академических часов).

(1 зачетная единица соответствует 36 академическим часам)

Видами учебных занятий и работы обучающегося по дисциплине являются:

Обозначение	Виды учебных занятий и работы обучающегося
Лаб	Лабораторные работы
СР	Самостоятельная работа обучающегося в период теоретического обучения
Контроль	Самостоятельная работа обучающегося и контактная работа обучающегося с преподавателем в период промежуточной аттестации

Структура дисциплины:

Форма обучения – очная.

№	Наименование раздела дисциплины	Семестр	Количество часов по видам учебных занятий и работы обучающегося						Формы промежуточной аттестации	
			Лек	Лаб	Пр	ОК	СР	Контроль		
1	Раздел I. Использование интегрированной среды разработки Eclipse для создания приложений Java Раздел II. Управляющие конструкции языка Java. Классы, объекты и методы Раздел III. Работа с массивами и исключениями	5		16	-			30	45	ПР-1 ПР-6
2	Раздел IV. Рекурсия, абстрактные классы и наследование	5		18						
3	Раздел VI. Разработка классов, обобщённые классы, коллекции, работа со строками, обработка исключений	6		36			30			
	Итого:			70		-	65	45		

I. СТРУКТУРА И СОДЕРЖАНИЕ ТЕОРЕТИЧЕСКОЙ ЧАСТИ КУРСА

Лекционные занятия не предусмотрены учебным планом. Студенты изучают теоретический материал самостоятельно.

Материалы теоретической части курса размещены в BlackBoard https://bb.dvfu.ru/webapps/blackboard/content/listContentEditable.jsp?content_id=102847_1&course_id=4130_1&mode=reset

Раздел I. Использование интегрированной среды разработки Eclipse для создания приложений Java

Тема 1. Компиляция программ в среде Eclipse. Классы объектов и классы-драйверы.

Компоненты среды Eclipse. Компоненты приложений Java. Компиляция приложений. Кодирование простых программ на Java. Различие между классами объектов и классами-драйверами. Создание класса объектов и класса-драйвера. Работа с классами Java API.

Тема 2. Встроенные типы данных и операции языка Java. Работа со строками String.

Типы данных в Java, примитивы, ссылочные типы. Объявление и инициализация переменных. Преобразование типов, апкастинг и даункастинг. Арифметические операции. Операции сравнения. Некоторые методы класса Math. Создание строк типа String. Конкатенация строк. Особенности работы со строками как со ссылочным типом. Использование методов compareTo() и equals(). Некоторые методы класса String.

Раздел II. Управляющие конструкции языка Java. Классы, объекты и методы

Тема 3. Использование сканера для ввода данных. Условный оператор.

Использование объекта класса Scanner для ввода данных. Условный оператор. Оператор множественного выбора. Оператор break. Условная операция.

Тема 4. Управляющие конструкции языка Java: операторы циклов.

Операторы циклов while, do..while, for.

Раздел III. Работа с массивами и исключениями

Тема 5. Работа с массивами. Сортировка и поиск

Одномерные массивы примитивных и ссылочных типов. Двумерные массивы примитивных и ссылочных типов. Объявление, инициализация массивов, обращение к элементам массивов. Альтернативные синтаксисы объявления массивов. Сущность задачи сортировки. Простейшие алгоритмы

сортировки массивов. Сравнение различных алгоритмов сортировки и поиска, эффективность, сложность алгоритмов сортировки и поиска.

Тема 6. Обработка исключений.

Различные типы ошибок в программах Java и приёмы работы с ними. Использование исключений. Перехват исключений. Передача исключений.

Раздел IV. Рекурсия, абстрактные классы и наследование

Тема 7. Создание классов, объектов и методов

Базовый шаблон класса. Создание объектов класса. Создание методов класса. Возвращаемые значения методов класса. Параметры методов класса. Конструктор. Оператор new. Сборщик мусора и финалайзер. Ссылка this. Инициализация данных класса в конструкторе.

Тема 8. Передача объектов в параметрах и перегруженные методы.

Модификаторы доступа. Передача объектов в методы и возврат объектов из методов. Методы с переменным числом параметров. Перегруженные конструкторы. Перегруженные методы.

Тема 9. Рекурсия, статические данные и методы, вложенные классы

Создание и использование статических данных. Создание и использование статических методов. Создание и использование статических классов. Методы с линейной и нелинейной рекурсией. Достоинства и недостатки рекурсии.

Тема 10. Наследование и полиморфизм.

Сущность наследования в программировании. Наследование классов Java. Подклассы и суперклассы. Доступ к методам суперкласса. Многоуровневая иерархия классов. Пример наследования с использованием апплетов. Переопределение методов. Динамический вызов переопределённых методов. Абстрактные методы и классы. Модификатор final. Класс Object.

Раздел V. Работа с приложениями Java

Тема 11. Развёртывание приложений

Работа с пакетами. Развёртывание приложений с использованием технологий Java Plug-in и Java Web Start. Двух и трёхслойная архитектура приложений Java.

Тема 12. Работа с готовым кодом Java.

Приемы чтения и анализа Java программ. Анализ программ с наследованием. Тестирование классов. Использование классов Array и ArrayList.

Раздел VI. Разработка классов, обобщённые классы, коллекции, работа со строками, обработка исключений

Тема 13. Разработка классов Java

Моделирование задачи с использованием классов Java. Приёмы, как сделать класс неизменяемым. Спецификаторы доступа public, private, protected и по умолчанию. Оператор instanceof. Виртуальные методы. Преобразование типов «вверх» и «вниз».

Тема 14. Обобщённые классы и коллекции значений.

Разработка обобщённых классов, их назначение. Коллекции с использованием и без использования обобщённых классов. Базовые коллекции – структуры данных: ArrayList, Set, HashMap. Реализация стека и очереди. Перечислимые типы.

Тема 15. Работа со строками.

Чтение строк. Поиск в строках. Парсинг строк. Работа с классом StringBuilder. Использование регулярных выражений для поиска, парсинга и замены в строках.

Тема 16. Обработка исключений.

Использование исключений для разработки надёжных приложений. Ключевые слова try и throw. Оператор catch. Оператор finally. Множественное исключение. Исключение с параметром. Использование assertion.

Тема 17. Ввод и вывод в Java.

Основы ввода и вывода в Java. Потоки ввода и вывода. Чтение данных с консоли, вывод данных на консоль. Использование потоков для чтения и записи файлов. Чтение и запись объектов с использованием сериализации.

II. СТРУКТУРА И СОДЕРЖАНИЕ ПРАКТИЧЕСКОЙ ЧАСТИ КУРСА И САМОСТОЯТЕЛЬНОЙ РАБОТЫ

Лабораторные работы (70 час)

Лабораторная работа №1	Работа со средой Eclipse. Работа со строками (8 часов)
Лабораторная работа №2	Условные операторы и циклы. (8 часов)
Лабораторная работа №3	Массивы. Сортировка и поиск (8 часов)
Лабораторная работа №4	Абстрактные классы и наследование (16 часов)

- Лабораторная работа №5** Развертывание приложений Java (12 часов)
- Лабораторная работа №6** Обобщенные классы, коллекции (10 часов).
- Лабораторная работа №7** Обработка исключений. (8 часов)

Лабораторные работы выполняются методом командной работы. При этом студенты разбиваются на команды по 3-5 человек, в каждой команде назначается руководитель (Team leader). Каждый член команды должен выполнить самостоятельно своё задание и выложить его на ветку в форуме среды Black Board по курсу. Руководитель команды даёт оценку практической работе своих партнеров по команде. Разделение на команды происходит во время практических занятий.

Срок на выполнение практической работы - одна неделя. Срок на оценку программы партнера по команде - одна неделя.

Преподаватель оценивает качество выполнения студентами своей практической работы и качество критики руководителем работ своих партнеров по команде. Срок выполнения работы также учитывается.

Задания для самостоятельной работы

Материалы для лабораторных работ размещены на учебном портале academy.oracle.com. Студентам необходимо получить у преподавателя доступ к своему личному кабинету.

III. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ

Учебно-методическое обеспечение самостоятельной работы обучающихся по дисциплине включает в себя:

- план-график выполнения самостоятельной работы по дисциплине, в том числе примерные нормы времени на выполнение по каждому заданию;
- требования к представлению и оформлению результатов самостоятельной работы;
- критерии оценки выполнения самостоятельной работы.

План-график выполнения самостоятельной работы по дисциплине

№ п/п	Дата/сроки выполнения	Вид самостоятельной работы	Примерные нормы времени на выполнение	Форма контроля
1	В течение 5 семестра	Изучение теоретического материала подготовка к лабораторным работам 1-3	20	ПР-1, ПР-6
2	17-18 неделя	Подготовка к экзамену	45	экзамен
3	15-18 неделя	Работа над итоговым проектом	10	Защита проекта
4	В течение 6 семестра	Изучение теоретического материала подготовка к лабораторным работам 4-7	20	ПР-1, ПР-6
5	34-36 неделя	Работа над итоговым проектом	10	Защита проекта
	Итого:		110	

Требования к представлению и оформлению результатов самостоятельной работы

Самостоятельная работа студентов состоит в изучении презентационных материалов по каждой теме, выполнении заданий для самостоятельной работы и работе над итоговым проектом.

Конспекты презентационных материалов доступны студентам на портале BlackBoard и на учебном портале academy.oracle.com.

Задания для самостоятельной работы также доступны студентам на портале BlackBoard.

Каждый студент должен выполнить самостоятельно своё задание и выложить его на проверку, воспользовавшись сервисом центра оценок. Преподаватель даёт оценку выполненным заданиям. Оценки учитываются при подсчёте рейтинга студентов для выставления итоговой оценки за семестр. Нормативный срок на выполнение практической работы - 5 дней. Срок учитывается при выставлении оценок.

В рамках самостоятельной по курсу студенты создают итоговый проект, который выполняется методом командной работы. Студенты объединяются в

команды по 3-5 человек, самостоятельно придумывают идею своего проекта и разрабатывают его.

Критерии оценки выполнения самостоятельной работы

– 100-86 баллов выставляется, если содержание и составляющие части соответствуют выданному заданию. Продемонстрировано владение навыками подготовки документа по теме. Фактических ошибок, связанных с пониманием структуры и содержания задания нет.

– 85-76 - баллов выставляется, если при выполнении задания допущено не более одной ошибки. Продемонстрировано владение навыками подготовки документа по теме. Фактических ошибок, связанных с пониманием структуры и содержания задания нет.

– 75-61 балл выставляется, если при выполнении задания допущено не более двух ошибок. Продемонстрировано навыки подготовки документа по теме. Допущено не более 2 ошибок, связанных с пониманием структуры и содержания задания.

60-50 баллов - если структура и содержание задания не соответствуют требуемым

Шкала оценивания

Менее 60 баллов	незачтено	неудовлетворительно
От 61 до 75 баллов	зачтено	удовлетворительно
От 76 до 85 баллов	зачтено	хорошо
От 86 до 100 баллов	зачтено	отлично

IV. КОНТРОЛЬ ДОСТИЖЕНИЯ ЦЕЛЕЙ КУРСА

№ п/п	Контролируемые модули/ разделы / темы дисциплины	Код индикатора достижения компетенции	Результаты обучения	Оценочные средства - наименование	
				текущий контроль	промежуточная аттестация

1	<p>Раздел I. Использование интегрированной среды разработки Eclipse для создания приложений Java</p> <p>Раздел II. Управляющие конструкции языка Java. Классы, объекты и методы</p> <p>Раздел III. Работа с массивами и исключениями</p> <p>Раздел IV. Рекурсия, абстрактные классы и наследование</p>	<p>ПК-6.1. демонстрирует знание основ моделирования и формальных методов конструирования программного обеспечения.</p>	<p>Знает современные технологии проектирования и производства программного обеспечения</p> <p>Умеет правильно выбрать технологию производства программного обеспечения для конкретного проекта</p>	<p>ПР1 лабораторные работы и задания для самостоятельного выполнения</p>	<p>Экзамен, вопросы № 1-11</p>
		<p>ПК-6.2. использует формальные методы конструирования программного обеспечения.</p>	<p>Владеет навыками применения технологий производства программного обеспечения</p>		
			<p>Знает современные технологии проектирования и производства программного обеспечения</p>		
			<p>Умеет правильно выбрать технологию производства программного обеспечения для конкретного проекта</p>		
			<p>Владеет навыками применения технологий производства программного обеспечения</p>		
			<p>Знает современные</p>		

	технологии реализации, оценки качества и анализа эффективности программного обеспечения
ПК-6.3. применяет методы формализации и моделирования программного обеспечения	Умеет правильно выбрать технологию реализации, оценки качества и анализа эффективности программного обеспечения
	Владеет навыками применения технологий реализации, оценки качества и анализа эффективности программного обеспечения
	Знает основные концептуальные положения функционального, логического, объектно-ориентированного и визуального направлений программирования
ПК-10.1. демонстрирует знание современных технологий разработки ПО (структурное, объектно-ориентированное).	Умеет создавать программы в рамках функционального, логического, объектно-ориентированного и визуального направлений программирования

	ия
	<p>Владеет навыками создания программ в рамках функционального, логического, объектно-ориентированного и визуального направлений программирования</p>
	<p>Знает основные концептуальные положения функционального, логического, объектно-ориентированного и визуального направлений программирования</p>
<p>ПК-10.2. использует структурное и объектно-ориентированное проектирование при разработке ПО.</p>	<p>Умеет создавать программы в рамках функционального, логического, объектно-ориентированного и визуального направлений программирования</p>
	<p>Владеет навыками создания программ в рамках функционального, логического, объектно-ориентированного и визуального</p>

			направлений программирования		
			Знает основные концептуальные положения функционального, логического, объектно-ориентированного и визуального направлений программирования		
		ПК-10.3. применяет современные технологии разработки ПО	Умеет создавать программы в рамках функционального, логического, объектно-ориентированного и визуального направлений программирования		
			Владеет навыками создания программных систем в рамках функционального, логического, объектно-ориентированного и визуального направлений программирования		
			Знает современные технологии проектирования и производства программного обеспечения		
2	Раздел VI. Разработка классов,	ПК-12.1. демонстрирует знание	Знает методы организации	ПР1	Зачет,

	<p>обобщённые классы, коллекции, работа со строками, обработка исключений</p>	<p>стандартов и моделей жизненного цикла ПО</p>	<p>работы в коллективах разработчиков ПО и методы сопровождения ПО.</p>	<p>лабораторные работы и задания для самостоятельного выполнения</p>	<p>вопросы № 12-19</p>
			<p>Умеет использовать методы организации работы в коллективах разработчиков ПО и методы сопровождения ПО.</p>		
			<p>Владеет навыками коллективной разработки и использования ПО</p>		
		<p>ПК-12.2. использует модели жизненного цикла при создании ПО</p>	<p>Знает методы организации работы в профессиональной деятельности.</p>		
			<p>Умеет использовать методы организации работы в профессиональной деятельности.</p>		
			<p>Владеет навыками организации работы в профессиональной деятельности.</p>		
		<p>ПК-12.3. применяет стандарты и модели жизненного цикла ПО</p>	<p>Знает методы разработки и сопровождения ПО при создании программных средств</p>		
			<p>Умеет</p>		

		применять методы разработки и сопровождения ПО при создании программных средств	
		Владеет навыками создания программных средств и их сопровождения	

Типовые контрольные задания, методические материалы, определяющие процедуры оценивания знаний, умений и навыков и (или) опыта деятельности, а также качественные критерии оценивания, которые описывают уровень сформированности компетенций, представлены в разделе VIII.

V. СПИСОК УЧЕБНОЙ ЛИТЕРАТУРЫ И ИНФОРМАЦИОННО МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Основная литература

1. Свистунов А.Н. Построение распределенных программных систем на Java: учебное пособие. М.: Интернет-Университет Информационных Технологий: БИНОМ. Лаборатория знаний, 2014. 279с.
<https://lib.dvfu.ru:8443/lib/item?id=chamo:797993&theme=FEFU>
2. Лафоре Р. Структуры данных и алгоритмы Java. Санкт-Петербург: Питер, 2016. 701с. <https://lib.dvfu.ru:8443/lib/item?id=chamo:817797&theme=FEFU>
3. Васюткина И.А. Технология разработки объектно-ориентированных программ на JAVA [Электронный ресурс]: учебно-методическое пособие/ Васюткина И.А. - Электрон. текстовые данные. - Новосибирск: Новосибирский государственный технический университет, 2012. - 152 с. - Режим доступа: <http://www.iprbookshop.ru/45047>.
4. Соколова В.В. Разработка мобильных приложений [Электронный ресурс]: учебное пособие/ Соколова В.В. - Электрон. текстовые данные. - Томск: Томский политехнический университет, 2014. - 176 с. - Режим доступа: <http://www.iprbookshop.ru/34706>

Дополнительная литература
(печатные и электронные издания)

1. Горнаков С.Б. Программирование мобильных телефонов на Java 2 Micro Edition. [Электронный ресурс]: М.: ДМК Пресс, 2008. — 511 с. — Режим доступа: http://e.lanbook.com/books/element.php?pl1_id=1189
2. Молчанова Л.А., Прудникова Л.И. Java в примерах и задачах: учеб.-метод. пособие [для вузов]. Владивосток: Изд-во Тихоокеанского экономического университета. — 2011. — Режим доступа: <http://lib.dvfu.ru:8080/lib/item?id=chamo:359168&theme=FEFU>
3. Баженова И.Ю. Языки программирования: учебник для высшего профессионального образования. Под редакцией В.А. Сухомлина. М.: Академия. — 2012. — 358 с. — Режим доступа: <http://lib.dvfu.ru:8080/lib/item?id=chamo:668317&theme=FEFU>
4. Курняван Б. Программирование web приложений на языке Java. М.: Лори, 2009. — 880 с. — Режим доступа <http://lib.dvfu.ru:8080/lib/item?id=chamo:382737&theme=FEFU>
5. Хабибуллин, Самоучитель JAVA. - СПб: БХВ-Петербург. - 2001. Режим доступа: <http://lib.dvfu.ru:8080/lib/item?id=chamo:672412&theme=FEFU>
6. Головин И.Г., Волкова И.А. Языки и методы программирования: учебник для вузов. М.: Академия, 2012. — 304с. <https://lib.dvfu.ru:8443/lib/item?id=chamo:668426&theme=FEFU>
7. Schildt Н. Introducing JavaFX 8 Programming. – New York: Oracle Press. – 2015.
8. Schildt Н. Java. A Beginners Guide. – New York: Oracle Press. – 2018.
9. Schildt Н. Java. The Complete Reference. – New York: Oracle Press. – 2018.

**Перечень ресурсов информационно-телекоммуникационной сети
«Интернет»**

1. The Java Tutorials - [Электронный ресурс]. Режим доступа - <http://docs.oracle.com/javase/tutorial/index.html>
2. IBM developerWorks Россия: Технология Java - [Электронный ресурс]. Режим доступа <http://www.ibm.com/developerworks/ru/java/>

Перечень информационных технологий и программного обеспечения

При осуществлении образовательного процесса студентами и профессорско-преподавательским составом используется следующее программное обеспечение:

1. Microsoft Office (Access, Excel, PowerPoint, Word и т. д).
2. Open Office.
3. Интегрированные среды разработчика Eclipse, NetBeans, Visual Studio.
4. Система контроля версий Git.

При осуществлении образовательного процесса студентами и профессорско-преподавательским составом используются следующие информационно-справочные системы:

1. Научная электронная библиотека eLIBRARY.
2. Электронно-библиотечная система издательства «Лань».
3. Электронная библиотека "Консультант студента".
4. Электронно-библиотечная система IPRbooks.
5. Информационная система "ЕДИНОЕ ОКНО доступа к образовательным ресурсам".
6. Доступ к электронному заказу книг в библиотеке ДВФУ, доступ к нормативным документам ДВФУ, расписанию, рассылке писем.

Лабораторные занятия проводятся в специализированном компьютерном классе.

Профессиональные базы данных и информационные справочные системы

1. База данных Scopus <http://www.scopus.com/home.url>
2. База данных Web of Science <http://apps.webofknowledge.com/>
3. Научная электронная библиотека <https://www.elibrary.ru/>
4. Общероссийский математический портал Math-Net.Ru <http://www.mathnet.ru>
5. Электронная библиотека диссертаций Российской государственной библиотеки <http://diss.rsl.ru/>
6. Электронная библиотека Европейского математического общества <https://www.emis.de/>
7. Электронные базы данных EBSCO <http://search.ebscohost.com/>

VI. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Дисциплина изучается в следующих организационных формах: лабораторное занятие; самостоятельное изучение теоретического материала; самостоятельное выполнение индивидуального проекта; индивидуальные и групповые консультации.

Основной формой самостоятельной работы студента является изучение конспекта лекций, их дополнение рекомендованной литературой, выполнение проекта, а также активная работа на лабораторных занятиях.

Контроль за выполнением самостоятельной работы студента производится в виде контроля каждого этапа работы, отраженного в портале BlackBoard и защиты проекта.

Студент должен планировать график самостоятельной работы по дисциплине и придерживаться его.

Особенности проведения лабораторных работ по дисциплине (активные и интерактивные формы)

Лабораторные работы по дисциплине проводятся параллельно с самостоятельной работой студентов. Студенты представляют результаты лабораторной работы преподавателю и друг другу, при этом стараются максимально обменяться опытом.

Рекомендуется использовать различные возможности работы с литературой: фонды научной библиотеки ДВФУ и электронные библиотеки (<http://www.dvfu.ru/library/>), а также доступные для использования другие научно-библиотечные системы.

К промежуточной аттестации допускаются обучающиеся, выполнившие все задания (контрольные, самостоятельные), предусмотренные учебной программой дисциплины, посетившие не менее 85% аудиторных занятий.

VII. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Перечень материально-технического и программного обеспечения дисциплины приведен в таблице.

Материально-техническое и программное обеспечение дисциплины

Наименование специальных помещений и помещений для самостоятельной работы	Оснащенность специальных помещений и помещений для самостоятельной работы	Перечень лицензионного программного обеспечения. Реквизиты подтверждающего документа
<p>690922, Приморский край, г. Владивосток, остров Русский, полуостров Саперный, поселок Аякс, 10, корпус D, ауд. D 733, 733а. Учебная аудитория для проведения занятий лекционного типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации</p>	<p>Помещение укомплектовано специализированной учебной мебелью (посадочных мест – 13) Оборудование: ЖК-панель 47", Full HD, LG M4716 CCBA – 1 шт. Доска аудиторная, Моноблок Lenovo C360G-i34164G500UDK с лицензионными программами Microsoft Office 2013(13 шт.) и аудиовизуальным и средствами проектор Panasonic DLPProjectorPT-D2110XE</p>	<p>1С Предприятия8 (8.2), 7-Zip, ABBYY Lingvo12,Alice 3, Anaconda3,Autodesk,CodeBlocks,CorelDRAW X7,Dia,Directum4.8,DosBox-0.74,Farmanager,Firebird 2.5,FlameRobin,Foxit Reader,Free Pascal,Geany,Ghostscript,Git,Greenfoot,gsview,Inscapе0.91,Java,Java development Kit,Kaspersky,Lazarus,LibreOffice4.4,MatLab R2017b,Maxima 5.37.2,Microsoft Expression,Microsoft Office 2013,Microsoft Silverlight,Microsoft Silverlight 5SDK-русский,MicrosoftSistem Center,Microsoft Visial Studio 2012,MikTeX2.9,MySQL,NetBeans,Notepad++,Oracle VM VirtualBox,PascalABC.NET,PostgreSQL 9.4,PTC Mathcad,Putty,PyQt GPL v5.4.1 for Pythonv 3.4,Python2.7(3.4,3.6),QGIS Brighton,RStudio,SAM CoDeC Pack,SharePoint,Strawberry Perl,Tecnomatix,TeXnicCenter,TortoiseSVN,Unity2017.3.1f1,Veusz,Vi m8.1,Visual Paradigm CE,Visual Studio2013,Windows Kits,Windows Phone SDK8.1,Xilinx Design ToolsAcrobat ReaderDC,AdobeBridge CS3,AdobeDeviceCentralCS3,Adobe ExtendScript Toolkit 2,Adobe Photoshpe CS3,DVD-студия Windows,GoogleChrome,Internet Explorer,ITMOproctor,Mozilla Firefox, Visual Studio Installer,Windows Media Center, WinSCP,</p>

Для освоения дисциплины требуется наличие проектора, аудиторная доска, компьютер.

Для проведения учебных занятий по дисциплине, а также для организации самостоятельной работы студентам доступно оборудование и специализированные кабинеты, соответствующие действующим санитарным и противопожарным нормам, а также требованиям техники безопасности при проведении учебных и научно-производственных работ.

В целях обеспечения специальных условий обучения инвалидов и лиц с ограниченными возможностями здоровья в ДВФУ все здания оборудованы пандусами, лифтами, подъемниками, специализированными местами, оснащенными туалетными комнатами, табличками информационно-навигационной поддержки.

VIII. ФОНДЫ ОЦЕНОЧНЫХ СРЕДСТВ

Для дисциплины «Основы программирования на Java» используются следующие оценочные средства:

Письменные работы:

1. Тест (ПР-1)
2. Контрольная работа (ПР-2)

Письменные работы

Письменный ответ приучает к точности, лаконичности, связности изложения мысли. Письменная проверка используется во всех видах контроля и осуществляется как в аудиторной, так и во внеаудиторной работе.

Тест (ПР-1) - система стандартизированных заданий, позволяющая автоматизировать процедуру измерения уровня знаний и умений обучающегося.

Контрольная работа (ПР-2) - средство проверки умений применять полученные знания для решения задач определенного типа по теме или разделу.

Методические рекомендации, определяющие процедуры оценивания результатов освоения дисциплины

Оценочные средства для промежуточной аттестации

Промежуточная аттестация студентов по дисциплине проводится в соответствии с локальными нормативными актами ДВФУ и является обязательной. Форма отчётности по дисциплине – экзамен. Экзамен проводится в тестовой форме.

Методические указания по сдаче экзамена

Экзамен принимается ведущим преподавателем. При большом количестве групп у одного преподавателя или при большой численности потока по распоряжению заведующего кафедрой (заместителя директора по учебной и воспитательной работе) допускается привлечение в помощь ведущему преподавателю других преподавателей. В первую очередь привлекаются преподаватели, которые проводили лабораторные занятия по дисциплине в группах.

В исключительных случаях, по согласованию с заместителем директора Школы по учебной и воспитательной работе, заведующий кафедрой имеет право принять экзамен в отсутствие ведущего преподавателя.

Форма проведения экзамена утверждается на заседании кафедры по согласованию с руководителем в соответствии с рабочей программой дисциплины.

Во время проведения зачета студенты могут пользоваться рабочей программой дисциплины, а также с разрешения преподавателя, справочной литературой и другими пособиями (учебниками, учебными пособиями, рекомендованной литературой и т.п.).

Время, предоставляемое студенту на прохождение теста на экзамене, должно составлять не более 60 минут.

Присутствие на зачете посторонних лиц (кроме лиц, осуществляющих проверку) без разрешения соответствующих лиц (ректора либо проректора по учебной и воспитательной работе, директора Школы, руководителя ОПОП или заведующего кафедрой), не допускается. Инвалиды и лица с ограниченными возможностями здоровья, не имеющие возможности самостоятельного передвижения, допускаются зачет с сопровождающими.

Вопросы к зачету и экзамену

Экзамен проводится в тестовой форме. Вопросы расположены на учебном портале ORACLE, доступны для зарегистрированных студентов.

Ниже приведены примеры вопросов к экзамену.

1 Method overloading is one of the ways that Java supports _____

Encapsulation

Class

Inheritance

* Polymorphism

2 Methods that have a return type other than void return a value to the calling routine using the following form of the return statement: return value;

* True

False

3 When we create a class, we are creating a new data type

* True

False

4 this can be used inside any method to refer to the current object

* True

False

5 A variable declared as final prevents its contents from being modified

* True

False

6 The new operator dynamically allocates memory for an object

* True

False

7 A parameter is a variable defined by a method that receives a value when the method is called.

*True

False

8 If the method does not return a value, its return type must be void

*True

False

9 The Code in java is contained within Methods

*True

False

10 The new operator dynamically allocates _____ for an object and returns a reference to it.

Classes

Variables

*Memory

None of the Above

11 The variable receiving the value returned by a method must also be compatible with the return type specified for the method.

*True

False

12 Deallocation of memory in Java is called Garbage Collection

*True

False

13 Static and Non-Static are the two types of nested classes

*True

False

14 Through _____ , you can control what parts of a program can access the members of a class

*Encapsulation

Class

Inheritance

Polymorphism

15 Java allows objects to initialize themselves when they are created using _____

Arguments

Classes

*Constructors

Parameters

16 A class is a template for an object, and an object is an instance of a class.

*True

False

17 When an overloaded method is invoked, Java uses the type and/or number of arguments as its guide to determine which version of the overloaded method to actually call

*True

False

18 Sometimes a method will need to refer to the object that invoked it. To allow this, Java defines the _____ keyword

* this
that
the
and

19 When a member is declared static, it CANNOT be accessed before any objects of its class are created, and without reference to any object

True

* False

Критерии выставления оценки студенту на экзамене

К экзамену допускаются обучающиеся, выполнившие программу обучения по дисциплине, прошедшие все этапы текущей аттестации.

Баллы	Оценка зачета/ экзамена	Требования к сформированным компетенциям
86-100	«зачтено»/ «отлично»	Оценка «отлично» выставляется студенту, если он глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, четко и логически стройно его излагает, умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, правильно обосновывает принятое решение, владеет разносторонними навыками и приемами выполнения практических задач.
76-85	«зачтено»/ «хорошо»	Оценка «хорошо» выставляется студенту, если он твердо знает материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, правильно применяет теоретические положения при решении практических вопросов и задач, владеет необходимыми навыками и приемами их выполнения.
61-75	«зачтено»/ «удовлетворительно»	Оценка «удовлетворительно» выставляется студенту, если он имеет знания только основного материала, но не усвоил его деталей, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала, испытывает затруднения при выполнении практических

		работ.
0-60	«не зачтено»/ «неудовлетворительно»	Оценка «неудовлетворительно» выставляется студенту, который не знает значительной части программного материала, допускает существенные ошибки, неуверенно, с большими затруднениями выполняет практические работы. Как правило, оценка «неудовлетворительно» ставится студентам, которые не могут продолжить обучение без дополнительных занятий по соответствующей дисциплине.

Оценочные средства для текущей аттестации

Текущая аттестация студентов по дисциплине проводится в соответствии с локальными нормативными актами ДВФУ и является обязательной.

Текущая аттестация проводится в форме собеседования (УО-1) по оцениванию фактических результатов обучения студентов и осуществляется ведущим преподавателем.

Объектами оценивания выступают:

- учебная дисциплина (активность на занятиях, своевременность выполнения различных видов заданий, посещаемость всех видов занятий по аттестуемой дисциплине);
- степень усвоения теоретических знаний;
- уровень овладения практическими умениями и навыками по всем видам учебной работы;
- результаты самостоятельной работы.

Составляется календарный план контрольных мероприятий по дисциплине. Оценка посещаемости, активности обучающихся на занятиях, своевременность выполнения различных видов заданий ведётся на основе журнала, который ведёт преподаватель в течение учебного семестра.

Студенты получают индивидуальные задания. В процессе их выполнения должны быть разработаны: формальная постановка задачи, алгоритмы ее решения, написана программа на любом языке программирования, для которой созданы тесты. Правильность работы программы демонстрируется с помощью

созданных тестов. Преподаватель вправе задать свои значения входных данных.

Вопросы к устному опросу

1. Приложения Java создаются в рамках
 - а **объектно-ориентированного программирования**
 - б. процедурного программирования
 - в. функционального программирования

2. Программы на Java
 - а. **позволяют создавать самодокументируемый код**
 - б. не позволяют создавать самодокументируемый код
 - в. требуют наличие отдельной БД для хранения документации

3. Возможность создания несколько точек входа в Java программу предназначена, в том числе, для
 - а. более гибкой разработки требований
 - б. **более удобного модульного тестирования**
 - в. развертывания программы на множестве устройств

4. Внутреннее модульное тестирование Java программ разработчиками возможно реализовать в том числе путём
 - а. **создания нескольких точек входа в программу**
 - б. написания специальных автоматизированных тестов
 - в. написания самотестируемого кода

5. Проектирование Java программ предполагает прежде всего проектирование
 - а. модулей и процедур
 - б. процедур и функций
 - в. **классов и интерфейсов**

- б. Освобождение неиспользуемой памяти в Java программах происходит
 - а. **с помощью специального сборщика мусора, автоматически запускаемого виртуальной машиной**
 - б. с помощью деструкторов

в. путём вызова программой функций освобождения памяти

7. Объекты классов Java, созданных программистом, представляют собой

а. ссылки

б. данные

в. конструкторы

8. Программы на Java

а. включают понятие глобальной переменной

б. включают понятие регистровой переменной

в. включают понятие статического поля класса

Критерии оценивания

Баллы (рейтингов ой оценки)	Оценка зачета (стандартная)	Требования к сформированным компетенциям
86-100	«зачтено»/ «отлично»	Оценка «зачтено» / «отлично» выставляется студенту, если он глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, четко и логически стройно его излагает, умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, правильно обосновывает принятое решение, владеет разносторонними навыками и приемами выполнения практических задач.
76-85	«зачтено»/ «хорошо»	Оценка «зачтено» / «хорошо» выставляется студенту, если он твердо знает материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, правильно применяет теоретические положения при решении практических вопросов и задач, владеет необходимыми навыками и приемами их выполнения.
61-75	«зачтено»/ «удовлетворительно»	Оценка «зачтено» / «удовлетворительно» выставляется студенту, если он имеет знания только основного материала, но не усвоил его деталей, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала, испытывает затруднения при выполнении практических работ.

0-60	«не зачтено»/ «неудовлетво рительно»	Оценка «не зачтено» / «неудовлетворительно» выставляется студенту, который не знает значительной части программного материала, допускает существенные ошибки, неуверенно, с большими затруднениями выполняет практические работы. Как правило, оценка «неудовлетворительно» ставится студентам, которые не могут продолжить обучение без дополнительных занятий по соответствующей дисциплине.
------	--	--